Stegräknare

Programmering av inbyggda system

Dennis Bunne
Simon Karlsson
Johan Kämpe

Sammanfattning

Innehållsförteckning

Inledning

Syfte

Syftet med projektet var att ta fram en stegräknare som brukar sig utav batteri samt flera moduler på I²C kanalen vilket skulle minska dess storlek. Och göras lätt tillgänglig för användning vid löpning, promenader etc.

Initialt används en OLED-display för att visa antal steg. Stegen ska avläsas med hjälp av en accelerometer, eller alternativt med en vibrations sensor.

Stegräknaren kan senare utökas med hastighetsmätare, tidräkning, ESP-modul eller bluetooth-modul för att skicka data m.m, om funktionen med stegräkning fungerar korrekt.

Bakgrund

Stegräknare

En stegräknare är en, vanligtvis portabel, enhet som mäter hur många steg en människa tar. Den kan användas vid exempelvis löpning eller promenad. Eller för att se hur många steg en person tar under en dag. Stegräknare kan också kallas pedometer.

I²C

I²C, Inter-Integrated Circuit, är ett sätt att koppla enheter till moderkort, inbyggda system, mobiltelefoner eller andra enheter. Kommunikation sker via två ledare, SDA (Serial Data Line, datasignal) och SCL (Serial Clock, klocksignal). Flera I²C-enheter kan anslutas till samma SDA- och SCL-ledare.

Stegräknarens funktion

Stegräknaren fungerar genom att hämta accelerationsdata från våran MPU9250. Där vi läser av datan från X,Y,Z-axlarna. Efter att ha bestämt ett offsetvärde, vilket är ett värde som väljs bort från en dataström kan vi med de olika axlarna utse ifall ett steg har tagits eller ej.

Genomförande och resultat

Använd programvara

Vid utveckling av stegräknaren har följande programvara använts

Program	**Beskrivning**
Atmel Studio 7.0	IDE för att skriva kod och kompilera till Atmel MCU.
Git	Versionshanteringsverktyg
GitKraken	Grafiskt gränssnitt för Git
Fritzing	Program för att skapa kretsscheman
Microsoft Word	Dokumentering
AVR Dudess	Program för att ladda upp koden till MCU i form av en HEX-fil.
Autodesk Fusion 360	CAD-program för att skapa ett hölje för produkten

Använd hårdvara

För att bygga stegräknaren har följande hårdvara använts:

Artikel	Beskrivning
Atmel ATMega328-PU DIP-28N 8-bit MCU	Microprocessor.
SparkFun MPU-9250	Enhet som mäter acceleration i X, Y och Z-riktning. Samt gyroskop, temperatur och kompass-funktionalitet.
Micro OLED Display SSD1306 128x64	Display med upplösningen 128x64.
AVR pocket programmer	Enhet för att programmera Atmega-MCU
DIL-hållare 28-pin 0.3"	Sockel för Atmega-MCU
Resistor 4,7kΩ, 2 st	Pull up-resistorer för I²C-kanaler
Kristalloscillator 16 MHz	Kopplas till MCU för att öka klockfrekvensen från 1 MHz till 16 MHz
Kondensator 22pF, 2 st	Ger jämn ström till Oscillatorn
Litiumbatteri 18650	Strömförsörjning
Batterihållare för litiumbatteri 18650	
Grovekontakter 4-stift	För modularisering av I²C-komponenter

Utveckling av stegräknaren

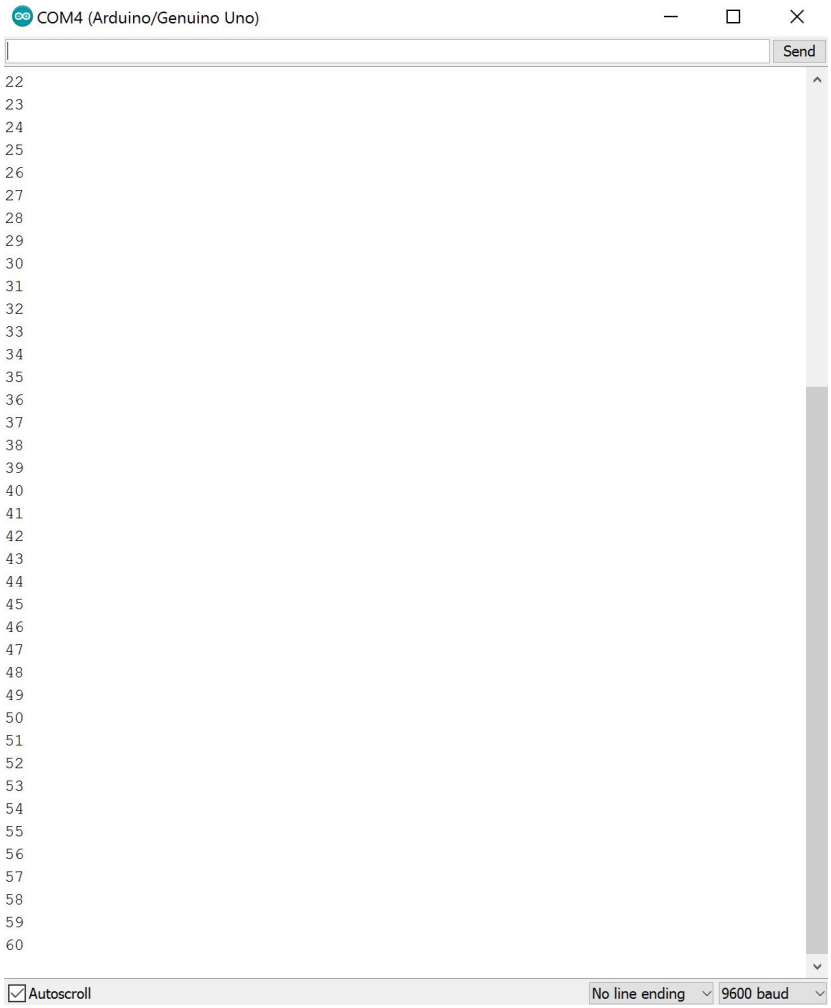
Två prototyper har byggts under projektets gång. Dessa beskrivs nedan.

Prototyp 1

Den första prototypen byggdes med en Arduino UNO och en breadboard dit komponenter kopplades. Prototypen testades genom att stoppa ner den i en ficka och promenerades runt med. Antal steg skrevs ut till *Serial Monitor* som finns i Arduinos IDE. En bärbar dator användes för strömförsörjning via USB, och för att se datan från Serial Monitor.

Både en accelerometer (MPU-9250) och vibrationssensor testades. Accelerometern valdes att gå vidare med, då denna ansågs ge bäst och stabilast värden.

Utskrift av steg till serial monitor:

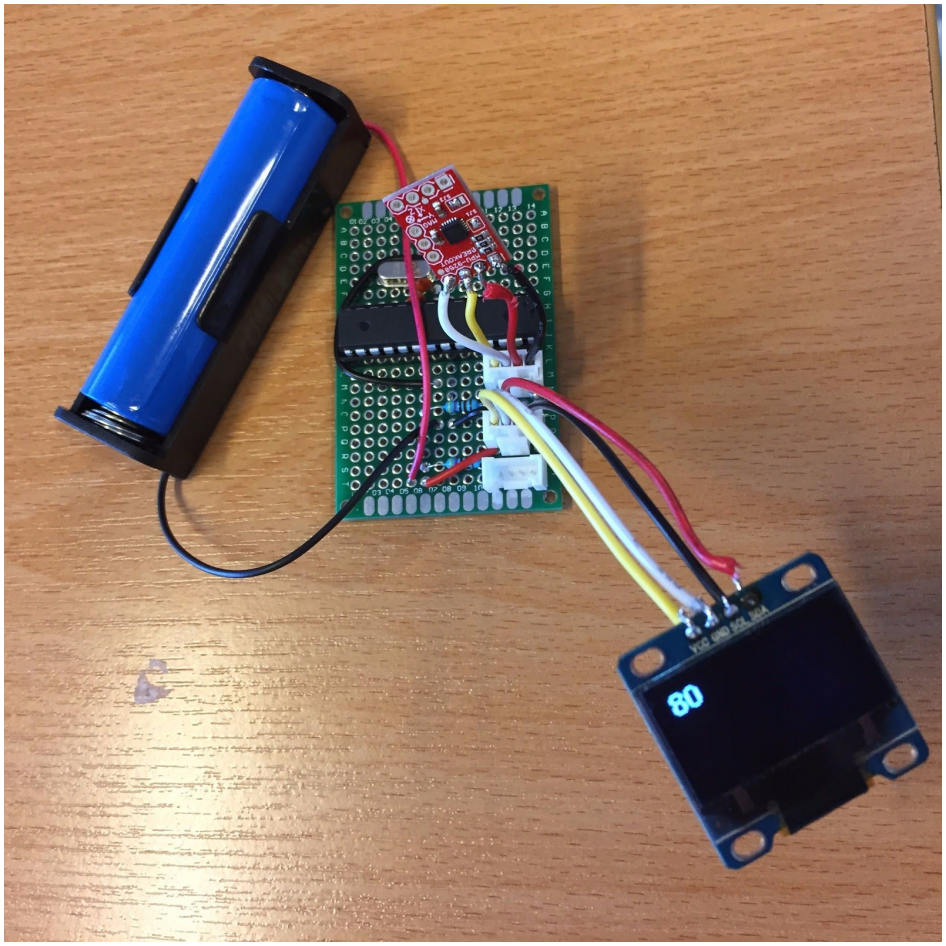


Prototyp 2

Till prototyp nummer 2 använd en Atmega328p MCU (Micro controller unit) i stället för Arduino. Komponenter har lödits samman på en platta och satts i ett hölje byggt av en gammal optisk mus.

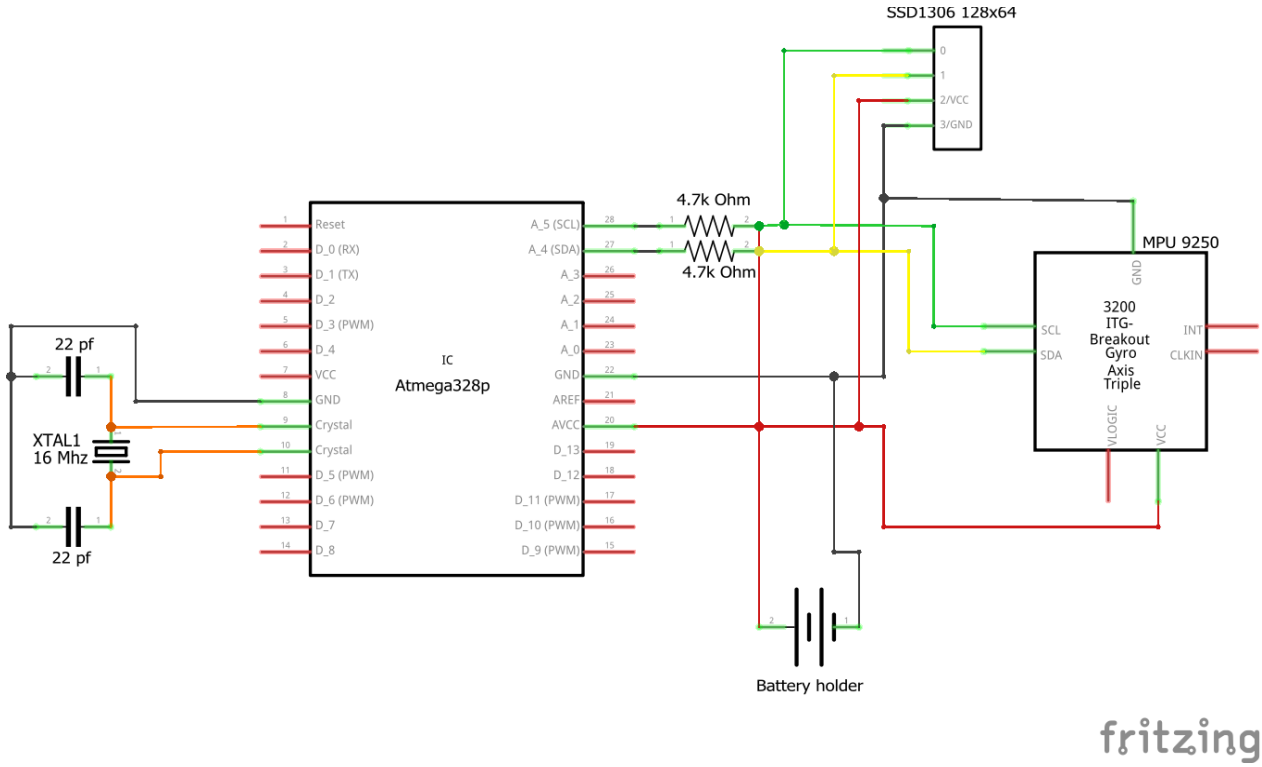
Kontakter har använts för modularisering. Samtliga I2C-enheter går att byta ut eller placera om. Ett batteri används för strömförsörjning. Stegräknaren har minskat i storlek avsevärt sedan prototyp 1.

Fotografi på prototyp 2, utan hölje:



Fotografi på prototyp 2, med hölje:

Kopplingschema för prototyp 2



Kod

Koden för stegräknaren är skrivet i programspråket C. Redan färdiga kodbibliotek för displayen, accelerometern och I2C har använts för att underlätta programmeringsarbetet.

Använda icke-standarbibliotek

Namn	Författare	Beskrivning
I2C master library using hardware TWI interface	Peter Fleury	I2C-funktionalitet för AVR-enheter (Atmel MCUer)
MPU6050 lib 0x02	Davide Gironi	För accelerometer-enheten, via I2C.
Universal 8bit Graphics Library	olikraus@gmail.com	För display, via I2C.

Globala variabler

Följande globala variabler används i projektets kod.

Namn	Beskrivning
uint16_t steps	Håller antal steg
double accIdle	Värde som sätts vid programmets uppstart, håller det värde som accelerometern ger när den ligger helt stilla.
double accCombined	Sammansatt accelerationsvärde från X- Y- och Z-riktning.
double accX, accY, accZ	Värden för acceleration i X- Y- och Z-riktning.

Definitioner / macron

Följande macron används i projektets kod

Namn	Värde	Beskrivning
X	**0x3B**	Adress för accelerometerdata i X-riktning i MPU
Y	**0x3D**	Adress för accelerometerdata i Y-riktning i MPU
Z	**0x3F**	Adress för accelerometerdata i Z-riktning i MPU
STEP_ACC_TRIGGER	**2.3**	BESKRIV!

Funktioner

Beskrivning av egenskrivna funktioner i projektets kod

Namn	Beskrivning
double getAcc(int addr);	Hämtar accelerationsdata från MPU och ger värdet i retur, angivet som g-kraft (SANT????). Som argument tar funktionen adressen till det register som ska läsas. Dessa adresser är definierade som macron **X, Y** och **Z.**
void drawSteps(uint16_t steps);	Skriver ut aktuellt antal steg till displayen.
void drawString(char * string);	Skriver ut en text till displayen. Texten ges som argument till funktionen, i form av en pekare till en char-sträng.
void setAccIdle();	Sätter ett värde till den globala variabeln **accIdle.** Funktionen körs endast en gång vid varje uppstart av programmet.
double getAccXYZ(void);	Använder sig av funktionen **getAcc** för att hämta accelerationsdata från **X-, Y-** och **Z**-riktningen. Det adderade värdet ges i retur.

Referenser

Länkar till författare av kodbibliotek

http://davidegironi.blogspot.se
https://github.com/olikraus
http://homepage.hispeed.ch/peterfleury

Länkar till använda datablad för komponenter

MPU9250
Atmega328p

Slutsats

TBD!!!